

## REMARKS/ARGUMENTS

The Examiner is again thanked for the performance of a thorough search. No claims have been added, cancelled, or amended. Hence, Claims 1-50 are pending in the application.

### THE REJECTIONS BASED ON THE PRIOR ART

#### Rejection Under 35 U.S.C. § 102(e)

Claims 39-49 were rejected under 35 U.S.C. § 102(e) as allegedly anticipated by Fernandez (U.S. Patent No. 6,785,673). The rejection is respectfully traversed.

**Claim 39** recites the following:

A method for processing XML data, comprising the computer-implemented steps of: receiving information at a first execution unit to cause said first execution unit to perform work associated with a query execution plan for servicing a request for data; wherein said information comprises an annotation that causes the XML data generated by said first execution unit to be transformed to a canonical form for use by a second execution unit; wherein said information, without said annotation, would cause said second execution unit to receive from said first execution unit XML data in a first form that cannot be used by said second execution unit; transforming XML data generated by said first execution unit to said canonical form prior to providing said XML data to said second execution unit; and providing XML data that is transformed to said second execution unit in said canonical form for use in performing work associated with said query execution plan by said second execution unit.

Such a method is neither taught nor suggested by Fernandez. For example, Claim 39 recites using a **query execution plan**, which one of ordinary skill in the art would recognize indicates that all of the processing that the claimed invention performs occurs within a database management system. By contrast, Fernandez describes a type of middleware that converts relational data to XML. It does not teach the use of a query execution plan.

The Office Action argues that FIGS. 6 and 7 illustrate the use of a query execution plan. The Office Action appears to equate the “query planner” of Fernandez to the query execution

plan recited in the claimed invention. Although their names are similar, there are significant differences between the Fernandez query planner and a query execution plan used in database management systems. The Applicants believe the Office Action confuses the operations that occur within a database management system with those that occur outside one.

For example, the Fernandez query planner “partitions a view tree into one or more subtrees” from which corresponding SQL queries are generated (col. 38, lines 17-18), and from which it is determined whether the cost of two separate queries is more or less than the cost of a single query combining the two separate SQL queries (col. 48, lines 21-32). In other words, the end result of the Fernandez process is a set of SQL queries that can be submitted to a relational database. This process and the set of SQL queries are not the same as a query execution plan. For example, the SQL queries created by Fernandez do not define the steps and operations performed by a database management server to process a query, as does a query execution plan. Stated otherwise, a query execution plan goes beyond simply generating SQL-type queries. A query execution plan details precisely how a database management system will execute one or more queries relative to the manner in which corresponding data is stored and indexed. These processes are not equivalent. Accordingly, the Applicants respectfully submit that Fernandez does not teach at least this element of Claim 39.

In addition, Fernandez does not teach or suggest the **use of multiple execution units for servicing a common data request**, e.g., according to a parallel query (PQ) framework implemented in a database server or across a distributed database, where each execution unit is a different process or thread. Fernandez does not show, teach, or suggest multiple execution units which work to service a common request. The Office Action cites FIG. 6 and/or Col. 37, lines 48-61 (it is unclear from the Office Action which section addresses the multiple execution

units) as teaching multiple execution units. However, neither cited section illustrates anything but a single execution unit.

For either cited section to map to the subject matter of Claim 1, **one or the other would also need to show that one process or thread generates a canonical form XML document and that a second process or thread uses the canonical form XML document to further perform part of the execution plan to further service the data request.** FIG. 6 clearly does not show such a multi-execution unit framework or architecture, nor does the remainder of Fernandez disclose such a multi-execution unit context. Consequently, Fernandez does not teach the use of multiple execution units in performing different portions of the execution plan. As the title illustrates, Fernandez is about converting relational data into XML, and is not about managing XML data among multiple execution units so that XML data from a particular execution unit, in a form that would not normally be compatible to a particular consumer, is made compatible to the consumer via the annotation-based transformation of the XML data output by the producer.

A further consequence of Fernandez not disclosing a multi-execution unit context is that **Fernandez does not teach that, but for the annotation, the producer would generate XML data in a canonical form.** The concept that but for the annotation, the first execution unit would generate XML data in a form that is unusable by the second unit. Fernandez discusses taking relational data and producing an XML document. In other words, Fernandez says nothing about an XML document being produced by one execution unit, converted into a canonical form, and then consumed by another execution unit while performing the same execution plan. Moreover, Fernandez says nothing about the form of the XML document produced by the first execution unit being unusable by the second execution unit.

Finally, the Claim 1 limitation about the annotation that causes the XML data generated by the first execution unit to be transformed to a canonical form for use by the second execution unit, the Office Action relies on a passage from Fernandez (col. 28, lines 1-5) which discusses a “compose” function that decomposes blocks of an XML-QL user query and “rewrites each nested pattern in a canonical form as a list of unnested patterns.” Applicants concede that Fernandez discusses transformation of XML-like data (a pattern) into a so-called canonical form. However, **the canonical form of Fernandez is simply a decomposition of a nested pattern into unnested patterns.** This says nothing about the multi-execution unit context of the subject matter of Claim 1 and nothing about the purpose of the transformation being to **transform XML data from a form that is unusable to a second execution unit into a form that is usable to the second execution unit**, where both the first execution unit and the second execution unit are working on portions of the same query execution plan.

Collectively, the foregoing distinctions between Claim 39 and the Fernandez reference illustrate how the disclosure of Fernandez is not a fair teaching of the subject matter recited in Claim 39. Hence, the disclosure of Fernandez cannot be stretched to form a valid anticipation rejection of Claim 39 and, therefore, Claim 39 is patentable thereover.

The distinctions between Claim 1 and Fernandez that are presented above are, generally, also applicable to the subject matter of independent **Claim 45**. That is, due to the deficiencies in the teachings of Fernandez regarding (a) a true query execution plan, (b) a multi-execution unit environment, and (c) transforming XML data generated by a first execution unit from a form that is unusable to a second execution unit into a form that is usable to the second execution unit, Fernandez does not support a valid anticipation rejection of this claims.

Dependent **Claims 40-44** depend from Claim 39 and dependent **Claims 46-49** depend from Claim 45. Therefore, each of these dependent claims is patentable over the cited reference

of record for at least the same reasons as the claim from which it depends. Furthermore, each of these dependent claims includes at least one other limitation that makes it further patentable over the reference of record. However, due to the fundamental differences between Claim 39 and Fernandez discussed above, discussion of these additional differences is unnecessary and is foregone at this time. The rejection of the dependent claims is collectively traversed, and no statements of official notice, overarching allegations of anticipation, or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed.

Based on the foregoing remarks, reconsideration and withdrawal of the rejection of Claims 39-49 under 35 U.S.C. § 102(e) is requested.

Rejection Under 35 U.S.C. § 103(a)

Claims 1-38 and 50 were rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Fernandez in view of Murthy et al. (U.S. Patent No. 7,103,590). The rejection is respectfully traversed.

First, independent Claims 1 and 50 recite limitations that are similar to some of those in Claim 39. For example, both Claims 1 and 50 recite (a) a query execution plan, (b) a multi-execution unit environment, and (c) transforming XML data generated by a first execution unit from a form that is unusable to a second execution unit into a form that is usable to the second execution limit. Thus, as argued above, Fernandez fails to teach or suggest elements of these claims. Murthy also fails to disclose or suggest these elements (and the Examiner has not asserted that Murthy does disclose these elements). Thus, even if the two references were combined (assuming *arguendo* that it would have been obvious to combine the references), the

combination still would not produce the method or system recited in Claims 1 and 50, respectively.

Moreover, even if the Fernandez and Murthy did teach the elements of Claims 1 and 50, still, the obviousness rejection is improper. There is no motivation or suggestion to combine the two references. As discussed above, Fernandez describes techniques for converting XML type queries into SQL queries that are submitted to a relational database. In other words, Fernandez is middleware that works outside of a database system to create properly structured relational queries. Murthy on the other hand describes processes that a database management system uses to pipeline data from one function to another. In other words, Murthy describes operations that occur within a database management system to speed up the querying process. One of ordinary skill in the art would realize that the two systems are distinct and that the operations performed by Murthy occur in a completely different context and in a completely different way than the query conversion of Fernandez. Again, a clear distinction can be brought up between those operations that occur within the database management system (e.g., those that affect query execution) and those that occur outside the database system (e.g., that affect the way a query is created and submitted to the database management system).

Thus, for at least these reasons, the Applicant submits that Claim 1 and 50 are patentable over Fernandez and Murthy, taken individually or in an alleged combination thereof. The Applicant further submits that claims 2-19, which depend from Claim 1 and which recite further advantageous aspects of the invention, are likewise patentable over Fernandez and Murthy for at least the reasons given above in connection with Claim 1.

Claims 20-38 are computer-readable storage medium claims analogous to Claims 1-19 and are patentable for at least the same reasons as Claims 1-19. Thus, the rejection of the dependent claims is collectively traversed, and no statements of official notice, overarching

allegations of anticipation, or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed.

**CONCLUSION**

For the reasons set forth above, it is respectfully submitted that all of the pending claims are now in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

/BrianDHickman#35894/

Brian D. Hickman

Reg. No. 35,894

2055 Gateway Place, Suite 550  
San Jose, CA 95110  
(408) 414-1080

**Date: August 23, 2007**

Facsimile: (408) 414-1076